

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

Applicants: S. HIRAGA, et al

Serial No.: 10/602,621

Filing Date: June 25, 2003

For: METHOD AND APPARATUS FOR MANAGING A DATABASE AND  
PROCESSING PROGRAM THEREFOR

Attention: MS Missing Parts

**LETTER CLAIMING RIGHT OF PRIORITY**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

September 17, 2003

Sir:


Under the provisions of 35 USC 119 and 37 CFR 1.55, applicants hereby claim  
the right of priority based on:

**Japanese Application No. 2002-188931  
Filed: June 28, 2002**

**Japanese Application No. 2003-177458  
Filed June 23, 2003**

Certified copies of said application documents are attached hereto.

Respectfully submitted,

  
\_\_\_\_\_  
Carl I. Brundidge  
Registration No. 29,621  
ANTONELLI, TERRY, STOUT & KRAUS, LLP

CIB/jdc  
Enclosures  
703/312-6600

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日            2 0 0 3 年    6 月 2 3 日  
Date of Application:

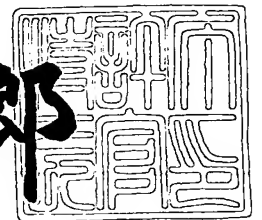
出 願 番 号            特 願 2 0 0 3 - 1 7 7 4 5 8  
Application Number:  
[ST. 10/C]:            [ J P 2 0 0 3 - 1 7 7 4 5 8 ]

出      願      人            株式会社日立製作所  
Applicant(s):            日立中国ソフトウェア株式会社

2 0 0 3 年    7 月 1 0 日

特許庁長官  
Commissioner,  
Japan Patent Office

太田信一郎



出証番号    出証特 2 0 0 3 - 3 0 5 6 2 6 4

【書類名】 特許願  
【整理番号】 K03003411A  
【あて先】 特許庁長官殿  
【国際特許分類】 G06F 12/00

## 【発明者】

【住所又は居所】 神奈川県横浜市戸塚区戸塚町 5 0 3 0 番地 株式会社日立製作所 ソフトウェア事業部内

【氏名】 板谷 孝

## 【発明者】

【住所又は居所】 広島県広島市中区銀山町 3 番 1 号 日立中国ソフトウェア株式会社内

【氏名】 平賀 茂人

## 【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所

## 【特許出願人】

【識別番号】 391002395

【氏名又は名称】 日立中国ソフトウェア株式会社

## 【代理人】

【識別番号】 100075096

【弁理士】

【氏名又は名称】 作田 康夫

## 【先の出願に基づく優先権主張】

【出願番号】 特願2002-188931

【出願日】 平成14年 6月28日

## 【手数料の表示】

【予納台帳番号】 013088

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9902691

【包括委任状番号】 9107836

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データベース管理方法および装置並びにその処理プログラム

【特許請求の範囲】

【請求項 1】

データベースを特定する識別情報とデータベースのアクセス属性情報を含むデータベース定義要求を入力し、該アクセス属性情報が改竄防止属性である場合は、当該データベースにおけるデータ挿入およびデータ参照のアクセスを許可し、少なくともデータ更新を禁止する属性を当該データベースに設定し、該設定後は該属性を変更不可とすることを特徴とするデータベース管理方法。

【請求項 2】

上記アクセス情報は削除禁止期間を含み、削除対象データの登録された時刻から該削除禁止期間が経過している場合は該削除対象データの削除を許可し、該削除禁止期間経過していない場合は該削除対象データの削除を禁止することを特徴とする請求項 1 記載のデータベース管理方法。

【請求項 3】

当該データベースに更にデータ削除を禁止することを特徴とする請求項 1 記載のデータベース管理方法。

【請求項 4】

当該データベースに更に少なくともデータベース名称の変更、データベースの削除、改竄防止属性の解除および削除禁止期間の変更のいずれかを禁止することを特徴とする請求項 1 記載のデータベース管理方法。

【請求項 5】

上記改竄防止属性を有するデータベースはアンロード日時をデータベースに保持し、該データベースをリロードする場合、外部記憶媒体に記憶されたアンロード日時と一致するか否かを判定し、一致する場合はリロードすることを特徴とする請求項 1 記載のデータベース管理方法。

【請求項 6】

アンロードする対象となるデータベースに改竄防止属性が設定されていない場合、改竄防止属性が設定されているデータベースへのリロードを禁止することを

特徴とする請求項 1 記載のデータベース管理方法。

【請求項 7】

上記設定されたデータベースのアクセス制御は、データベース管理者を含めた全てのユーザのアクセスに対して適用することを特徴とする請求項 1 から 6 記載のいずれかのデータベース管理方法。

【請求項 8】

データベースを特定する識別情報とデータベースのアクセス属性情報を含むデータベース定義要求を入力する手段と、該アクセス属性情報が改竄防止属性であり、該データベース定義要求が最初の改竄防止属性の設定である場合は、当該データベースにおけるデータ挿入およびデータ参照のアクセスを許可し、少なくともデータ更新を禁止する属性を当該データベースに設定する手段とを備えたことを特徴とするデータベース管理装置。

【請求項 9】

データベースを特定する識別情報とデータベースのアクセス属性情報を含むデータベース定義要求を入力するステップと、該アクセス属性情報が改竄防止属性である場合は、当該データベースにおけるデータ挿入およびデータ参照のアクセスを許可し、少なくともデータ更新を禁止する属性を当該データベースに設定するステップと、該設定後は該属性を変更不可とするステップとを有することを特徴とするデータベース管理プログラム。

【請求項 1 0】

上記アクセス情報はデータ更新猶予期間を含み、該データ更新猶予期間の設定されたデータが挿入れた時刻から該データ更新猶予期間が経過していない場合は該登録対象データの更新を許可し、該データ更新猶予期間が経過した場合は該データの更新を禁止することを特徴とする請求項 1 記載のデータベース管理方法。

【請求項 1 1】

上記改竄防止属性は、当該データベースにおける列に設定することを特徴とする請求項 1 記載のデータベース管理方法。

【発明の詳細な説明】

【0 0 0 1】

**【発明の属する技術分野】**

本発明はデータベースのアクセス権限を管理するデータベース管理技術に関する。

**【0002】****【従来の技術】**

データベースのアクセス権限を管理するデータベース管理システムにおいて、データベースへのアクセス権限は、データベース所有者及びデータベース所有者からアクセス権限を許可された者が持つことができ、テーブル毎やユーザ毎のデータベースへのアクセス権限の設定や管理を行うためには極めて有用である。

**【0003】**

しかし、更新を不可としたいデータベースに対して、所有者や更新権限を許可された者またはこれらの者になりすました者が、不正にまたは誤って更新されてしまうことがある。分散データベースにおけるデータベースへの不正なアクセスを防止しようとする技術として、特許文献1のような技術が開示されている。

**【0004】****【特許文献1】**

米国特許5,924,094号

**【0005】****【発明が解決しようとする課題】**

従来の技術においては、更新を不可としたいデータに対して、表所有者や更新権限を譲渡された者またはこれらの者になりすました者が、不正にまたは誤って更新することを防ぐことはできないという課題があった。

**【0006】**

本発明の目的は、改竄防止属性が設定された場合、不正にまたは誤ってデータベースへの処理が実行されることを防ぐことにある。

**【0007】****【課題を解決するための手段】**

本発明は、データベース定義時に改竄防止属性を指定した場合に、データベースの属性としてディクショナリに登録し、データ更新要求時にデータベースの属

性をチェックし、当該データベースが改竄防止属性有りの場合は、データベースの所有者であっても表の変更を不可とする。これにより、不正にまたは誤って表データを更新することを防ぐことが可能となる。

#### 【0 0 0 8】

また、データベース定義時に改竄防止属性に加え行削除禁止期間及び行挿入日時保持列名を指定した場合、これらの情報をデータベースの属性としてディクショナリに登録し、行挿入時に行挿入日時保持列にデータベース管理システムが行挿入日時を設定し、データベース所有者またはデータベース所有者から表更新権限を譲渡された者であっても当該列データを更新不可とする。そして、行削除要求時に表の属性及び行挿入日時をチェックし、当該データベースが改竄防止属性を持ち、かつ、行削除禁止期間を指定されている場合は、行挿入後、行削除禁止期間を経過するまでの間は、データベース所有者であっても行の削除を不可とすることが可能となる。これにより、データが挿入された日より一定の期間すなわち行削除禁止期間で指定した期間、当該データの削除を禁止することが可能となる。

#### 【0 0 0 9】

さらに、データベースのアクセス制御情報が格納されているディクショナリに登録した改竄防止属性情報を参照することで、データベース定義削除又はデータベース定義変更によるデータの不正削除・更新を防ぐことが可能となる。

#### 【0 0 1 0】

さらに、アンロード日時をステータスファイル及び外部記憶媒体に記録し、ディクショナリに登録した改竄防止属性情報を参照することで、リロード時に不正に改竄防止属性有りのデータを削除又は更新することを防ぐことが可能となる。

#### 【0 0 1 1】

#### 【発明の実施の形態】

以下、本発明の一実施例を図 1 から図 1 0 を参照しながら説明する。

#### 【0 0 1 2】

本発明に係わるデータベース管理方法及び装置を図 1 に示す。データベース管理システムは、CPU、メモリ、端末装置、ハードディスク装置等から構成され

る電子計算機システムにおいて、オペレーティングシステム上で動作し、論理的なデータベース格納領域及び表や格納領域の定義情報を格納するディクショナリは、ハードディスク装置上に割り当てられる。

#### 【 0 0 1 3 】

ディクショナリは、データベースのテーブル構造、列定義、インデクス定義などを含むデータベース設計仕様を格納するものであり、データベース参照・更新時にディクショナリの情報が必要となる。本発明の特徴となる改竄防止情報を管理するディクショナリ表は、表名列、改竄防止属性列、行削除禁止期間列、行挿入日時保持列名列から構成され、ディクショナリに格納される。このディクショナリ表は、図 2 における 2 0 2 に相当する。本発明はリレーショナル・データベースを一例に説明しているがリレーショナル・データベースに限定されることなく種々のデータベースに適用可能である。ここでは、改竄防止とは、データベース所有者およびデータベース所有者から権限を許可されたものについてもデータベースに登録済みのデータの変更を許可しないアクセス権限の機能をいう。この改竄防止属性は、データベース所有者およびデータベース所有者から権限を許可されたものでは変更不可能であり、更に上位のデータベース管理者のレベルでないと変更はできない。

#### 【 0 0 1 4 】

図 2 は、本発明の特長を表す表定義文の一例と、改竄防止定義情報がディクショナリに格納されている様子を示す図である。S Q L 1 に示す表 1 及び S Q L 2 に示す表 2 は、2 0 1 に示すパラメタ名 P 1 に Y を指定することで、改竄防止属性有りの表となり、ディクショナリ 2 0 2 の改竄防止属性列にその旨を示す「Y」が格納される。また、表 2 には改竄防止属性に加え、2 0 1 に示すパラメタ名 P 2 に行削除禁止期間を指定し、かつ 2 0 1 に示すパラメタ名 P 3 に行挿入日時保持列を指定することで、行挿入後、削除禁止期間を経過した行のみ削除可能となる表となり、ディクショナリの行削除禁止期間列及び行挿入日時保持列名列に各々指定された値が格納される。本例では、リレーショナルデータベースの例で説明したので行挿入、行削除の例を説明したが、データベース一般にはデータの挿入、削除についても同様にデータの挿入は許可するがデータの削除は禁止する

ことになる。なお、図 2 においてはパラメタ P 3 をユーザ設定とする例としているが、ユーザ設定とはせずシステムで暗黙的に自動設定とすることも可能である。

#### 【0015】

表データ更新不可・行削除不可の表を定義する場合の S Q L の一例が S Q L 1 である。S Q L 1 で定義される表 1 は、行挿入および検索（もしくは参照）は可能であるが、表データ更新および行削除は不可の表となる。一方、表データ更新不可である点は S Q L 1 と同様であるが、挿入後、一定期間を経過した行に限り削除可能となるオプションを指定した S Q L の一例が S Q L 2 である。この S Q L 例では、行挿入後、3 年を経過した行に限り削除可能となる。S Q L 2 で定義される表 2 は、行挿入・検索は可能、表データ更新は不可である点は表 1 と同様であるが、行挿入後、一定期間を経過した行については削除可能となる点が表 1 とは異なる。S Q L 2 の S Q L 例では、行削除禁止期間に「3 Y E A R S （3 年）」を指定し、行挿入履歴保持列名として「カルテ作成日」を指定している。これにより、表 2 に行挿入時、行挿入日時が「カルテ作成日」にシステムにより自動的に格納され、表 2 の行削除要求時、「カルテ作成日」に格納された行挿入日時に 3 年を加算した結果と行削除要求日時とを比較し、削除しようとする行が挿入後 3 年を経過した行であれば削除可能となる。データベースに設定されたこれらの属性は、当該データベースの所有者やその権限を許可されたものでは変更することができない。つまり、一度設定されると、当該データベースの所有者やその権限を許可されたものでは変更することができないことになる。このようにすることにより、データベースが不正にまたは誤って変更されることを防ぐ。

#### 【0016】

図 3 は、表に対するデータ挿入処理の流れを示す図である。表に対しデータを挿入する場合、まず改竄防止属性有りの表か否かをチェックし（3 0 1）、改竄防止属性無しの場合はそのままデータ挿入処理を実行する（3 0 4）。3 0 1 で改竄防止属性有りの表の場合は、行削除禁止期間の指定が有るか否かチェックし（3 0 2）、行削除禁止期間の指定が無い場合はデータ挿入処理を実行する。3 0 2 で行削除禁止期間の指定が有る場合は、行挿入履歴保持列に格納するデータ

(時刻印)を作成後、データ挿入処理を実行する。ここで、行挿入履歴保持列に格納するデータをユーザが指定することはできない。例え指定されていたとしても、指定値を無視し、システムが作成したデータを格納する。具体例としては、図4の表2は、「カルテID」列が「001」の行を2002年6月1日に挿入し、同様に「カルテID」列が「002」の行を2002年6月10日に挿入、「カルテID」列が「003」の行を2002年6月18日に挿入した場合の例である。

#### 【0017】

図2で定義した表1及び表2にデータ挿入後、表データの更新SQL及び行削除SQLを実行した場合の例を図4に示す。ここで、SQL1は表1のある行に対する更新を指示するSQL、SQL2は表1のある行の削除を指示するSQL、SQL3は表2のある行に対する更新を指示するSQL、SQL4は表2の「カルテID」列の値が「001」である行の削除を指示するSQLである。改竄防止属性有りの表1又は表2に対し、表データ更新を指示するSQL(SQL1又はSQL3)を実行した場合、データベース管理システムはUPDATE不可の旨を示す情報をUAPに返す(401)。行削除禁止期間指定無しの表1に対し、行削除を指示するSQL(SQL2)を実行した場合、データベース管理システムはDELETE不可の旨を示す情報をUAPに返す。行削除禁止期間有りの表2に対し、行削除を指示するSQL(SQL4)を実行した場合、削除対象の行が行削除禁止期間経過前であればデータベース管理システムはDELETE不可の旨をUAPに返し、削除対象の行が行削除禁止期間経過後であればデータベース管理システムはDELETE処理を実施する。

#### 【0018】

図5は、表データの更新処理の流れを示す図である。表データ更新要求があった場合、まず改竄防止属性有りの表か否かをチェックし(501)、改竄防止属性無しの場合は表データ更新実行者が表の所有者か否かをチェックし(502)、表の所有者であれば更新処理を実行する(504)。502で表の所有者でない場合は、表データ更新実行者が表の所有者から当該表に対する更新権限を譲渡されているかチェックし(503)、譲渡されていれば更新処理を実行し(504)

）、譲渡されていなければ更新不可とする（505）。一方、501で改竄防止属性有りの場合は、例え表の所有者であったとしても更新不可となる（506）。具体例として、図4のSQL1及びSQL3のSQL例では、501のチェックでYESとなり、更新不可となる。

#### 【0019】

図6は、行削除処理の流れを示す図である。行削除要求があった場合、まず改竄防止属性有りの表か否かをチェックし（601）、改竄防止属性無しの場合は行削除実行者が表の所有者か否かをチェックし（602）、表の所有者であれば行削除処理を実行する（604）。602で表の所有者でない場合は、行削除実行者が表の所有者から当該表に対する行削除権限を譲渡されているかチェックし（603）、譲渡されていれば更新処理を実行し（604）、譲渡されていなければ行削除不可とする（605）。一方、601で改竄防止属性有りの場合は、次に行削除禁止期間指定有りの表か否かをチェックし（606）、行削除禁止期間の指定が無しであれば、例え表の所有者であったとしても削除不可となる（608）。606で行削除禁止期間の指定が有りの場合は、削除対象の行中の行挿入日時保持列の値に行削除禁止期間列の値を加算し行削除実行日と比較することで、削除禁止期間を経過した行か否かをチェックし（607）、削除禁止期間を経過した行であれば602の処理に分岐し、削除禁止期間を経過していない行であれば行削除不可とする（608）。具体例として、図4のSQL2のSQL例では、601でYES、606でNOとなり、行削除不可となり、SQL3のSQL例では、601でYES、606でYES、607の判定で削除禁止期間を経過後の行ではYESとなり表所有者又は表所有者から削除権限を譲渡されている者であれば行削除可、削除禁止期間を経過前の行ではNOとなり行削除不可となる。

#### 【0020】

図7は、表定義の削除処理の流れを示す図である。表定義削除要求があった場合、まず改竄防止属性有りの表か否かをチェックし（701）、改竄防止属性無しの場合は表定義削除実行者が表の所有者又はDBA権限者か否かをチェックし（702）、表の所有者又はDBA権限者であれば表定義削除処理を実行する（

703)。ここでの「DBA権限者」とは、データベース管理システムにおける管理者権限を持つユーザのことである。通常、データベース管理システムにおいては、「DBA権限者」と「一般利用者」に大別され、DBA権限者は一般利用者には利用できないデータベースを管理する機能を利用することが可能である。702で表の所有者又はDBA権限者でない場合は、表定義削除不可となる(704)。一方、701で改竄防止属性有りの表の場合は、表にデータがあるか否かをチェックし(705)、表にデータが有る場合は、例えば表の所有者又はDBA権限者であったとしても表定義削除不可となる(706)。705で表にデータが1件も無い場合は、702に分岐する。具体例として、図4の表1・表2に対する表定義削除要求があった場合を考える。この場合、701でYES、705でYESとなり表定義削除不可となる。

#### 【0021】

図8は、表定義の変更処理の流れを示す図である。表定義変更要求があった場合、まず改竄防止属性有りの表か否かをチェックし(801)、改竄防止属性無しの場合は表定義変更実行者が表の所有者又はDBA権限者か否かをチェックし(802)、表の所有者又はDBA権限者であれば表定義変更処理を実行する(803)。802で表の所有者又はDBA権限者でない場合は、表定義変更不可となる(804)。一方、801で改竄防止属性有りの表の場合は、表名の変更、改竄防止属性の解除、行削除禁止期間の変更、又は行挿入日時保持列の削除のいずれかであるか否かをチェックし(805)、805のいずれかに合致する場合は、例えば表の所有者又はDBA権限者であったとしても表定義変更不可となる(806)。805でいずれの条件にも合致しない場合は、802に分岐する。具体例として、図4の表1・表2に対する表定義変更要求(表名の変更、改竄防止属性の解除、行削除禁止期間の変更、又は行挿入日時保持列の削除)があった場合を考える。この場合、801でYES、805でYESとなり表定義変更不可となる。

#### 【0022】

図9は、本発明における表データのアンロード・リロード処理を示す図である。ここで、ステータスファイルとは、データベース管理システムの状態やデータ

ベースの状態等の情報を格納し、データベース管理システムを再起動するときに参照するファイルのことである。また、アンロードとは、データベースの内容を、磁気テープ等の外部記憶媒体に複写することであり、バックアップとも呼ばれる。また、リロードとは、アンロードしたデータをデータベースに書き戻すことを言う。図 9 では、改竄防止属性有りの表（表 1 及び表 2）と改竄防止属性無しの表（表 3）をそれぞれ 2 0 0 2 年 6 月 2 0 日、同 2 5 日、同 1 8 日にアンロードし、その後の表 1 及び表 3 のアンロード媒体の各表へにおけるリロード可否について説明する。まず、改竄防止属性有りの表（表 1）のデータをアンロードする場合について説明する。アンロードユティリティ 9 0 1 は、改竄防止属性有りの表 1 のデータを外部記憶媒体 9 0 3 にコピーする。この時、アンロードユティリティは、表名及びアンロード日時をステータスファイル（9 0 5）及び外部記憶媒体 9 0 3 に記録する。9 0 3 をリロードユティリティ 9 0 2 でデータベースに書き戻す際、同一表（表 1）へのリロードは、ステータスファイル中のアンロード日時と 9 0 3 中のアンロード日時及び表名とが一致する場合にのみ実行できる。他の改竄防止属性有りの表（表 2）へのリロードは不可となる。改竄防止属性無しの他表（表 3）へのリロードは可である。次に、改竄防止属性無しの表（表 3）のデータをアンロードする場合について説明する。アンロードユティリティで表 3 のデータを 9 0 4 にコピーする。9 0 4 は、改竄防止属性無しの表に対してはリロード可能であるが、改竄防止属性有りの表（表 1 又は表 2）に対してはリロードできない。

### 【 0 0 2 3 】

図 1 0 は、リロード処理の流れを示す図である。リロード処理では、まずアンロード元が改竄防止属性有りの表か否かをチェックし（1 0 0 1）、アンロード元が改竄防止属性無しの表の場合は次にリロード先が改竄防止属性有りの表か否かをチェックし（1 0 0 2）、リロード先が改竄防止属性無しの表であればリロード処理を実行する（1 0 0 3）。1 0 0 2 で改竄防止属性有りの表の場合は、リロード不可（1 0 0 4）となる。1 0 0 1 で改竄防止属性有りの表の場合は、次にリロード先が改竄防止属性有りの表か否かをチェックし（1 0 0 5）、リロード先が改竄防止属性無しの表であればリロード処理を実行する（1 0 0 3）。

1 0 0 5 でリロード先が改竄防止属性有りの表であれば、次にアンロード元の表名とリロード先の表名が同じか否かチェックし (1 0 0 6)、表名が異なればリロード不可 (1 0 0 8) となる。1 0 0 6 で表名が同じ場合は、次にステータスファイル中の当該ファイルのアンロード日時と外部記憶媒体中のそれとを比較し (1 0 0 7)、一致する場合はリロード処理を実行し、不一致の場合はリロード不可 (1 0 0 8) となる。具体例として、図 9 の表 1 をアンロードした 9 0 3 を表 2 にリロードする場合を考える。アンロード元の表 1 は改竄防止属性有りの表なので 1 0 0 1 で Y E S、リロード先の表 2 も改竄防止属性有りの表なので 1 0 0 5 で Y E S、表 1 と表 2 は表名が異なるので 1 0 0 6 で N O となり、リロード不可となる。さらに具体例として、図 9 の表 1 をアンロードした 9 0 3 を表 1 自身にリロードする場合を考える。この場合は、1 0 0 1 で Y E S、1 0 0 5 で Y E S、1 0 0 6 で Y E S となり、9 0 3 中のアンロード日時とステータスファイル中の表 1 アンロード日時とが一致するため 1 0 0 7 で Y E S となりアンロード可となる。表 1 に対する古い世代のアンロード媒体を使用した場合、1 0 0 7 でアンロード日時が不一致となるため、アンロード不可となる。さらに具体例として、図 9 の表 3 をアンロードした 9 0 4 を表 1 にリロードする場合を考える。アンロード元の表 3 は改竄防止属性無しのため 1 0 0 1 で N O、リロード先の表 1 は改竄防止属性有りのため 1 0 0 2 で Y E S となり、リロード不可となる。

#### 【 0 0 2 4 】

以上述べたように処理することにより、ディクショナリに表の属性として改竄防止属性、行削除禁止期間、及び行挿入日時保持列名を持ち、表データ更新または行削除要求時に表の属性及び行挿入日時をチェックすることで、更新を不可としたい表データに対し、表所有者や更新権限を譲渡された者であっても更新を不可とし、不正にまたは誤って更新することを防ぐことを可能とし、また、行挿入後、一定期間行データを削除不可としたい表データに対し、表所有者や行削除権限を譲渡された者であっても行削除を不可とし、不正にまたは誤って行を削除することを防ぐことを可能とする。

#### 【 0 0 2 5 】

図 1 1 から図 1 4 は図 2 から図 8 と同様に改竄防止属性有りの表に対するデー

タ更新を不可とする実施例を示す図であるが、図 2 から図 8 と異なり、図 2 に示すパラメタ P 3 を利用者が指定をしなくともパラメタ P 1 の指定の有無により暗黙的に設定する実施例、およびデータ挿入日であればデータの更新を可能とする実施例、および列単位で改竄防止属性の定義を可能とする実施例である。

#### 【 0 0 2 6 】

図 1 1 は図 2 と同様に本発明の特長を表す表定義文の一例と、改竄防止定義情報がディクショナリに格納されている様子を示す図であり、パラメタ P 3 を利用者が指定をしなくともパラメタ P 1 の指定の有無により暗黙的に設定する一例および列単位で改竄防止属性の定義を可能とする一例である。S Q L 1 は、図 2 で示した表 2 と同じ属性を持ち、パラメタ P 3 を指定せずに定義する場合の S Q L の一例である。この S Q L 例では、行挿入後、3 年を経過した行に限り削除可能となる。S Q L 1 では、行削除禁止期間に「3 Y E A R S ( 3 年 ) 」を指定し、行挿入履歴保持列名は指定していない。D B 定義処理 1 1 0 4 は S Q L 中にパラメタ P 2 が指定され、かつパラメタ P 3 が指定されていない場合は、行挿入日時保持列を暗黙的に設定する必要があると判断し、行挿入保持列として列「X X X X X」を暗黙的に定義する。これにより、ディクショナリ 1 1 0 2 の表 3 に対応する行挿入日時保持列名列に「X X X X X」が格納される。本例では、D B 定義処理 1 1 0 4 が行挿入日時保持列名として「X X X X X」を定義する例を示したが、一般的なデータベース管理システムにおいては、利用者が任意に列名の変更や追加を行うことが可能である。従って、D B 定義処理 1 1 0 4 が行挿入日時保持列名を暗黙的に設定する際の列名称に関しては、利用者が指定できない名称にするなどの配慮が必要となる。S Q L 2 に示す表 4 は 1 1 0 1 に示すパラメタ名 P 1 を列単位に指定することにより、列単位で改竄防止属性の設定を可能とする例である。表 4 は「カルテ I D」に対し「Y」を指定することで、改竄防止属性有りの表でかつ、列「カルテ I D」は改竄防止属性有りの列となり、ディクショナリ 1 1 0 2 の改竄防止属性列にその旨を示す「Y」が格納され、さらに列の定義情報を管理するディクショナリ表 1 1 0 3 の改竄防止属性列にその旨を示す「Y」が格納される。また、表 4 は改竄防止属性に加え、1 1 0 1 に示すパラメタ名 P 2 に行削除禁止期間が指定されているが、行削除禁止期間の定義の流れにつ

いては表 3 と同じであるため説明は省略する。

### 【 0 0 2 7 】

図 1 2 は図 1 1 で定義した列単位に改竄防止属性を定義した表 4 にデータ挿入後、表データの更新 S Q L を実行した場合の例を示し、さらに改竄防止属性有りの表あるいは改竄防止表あり列に対する表データ更新を行挿入日当日であれば可能とする例を示す。図 2 においては、表全体を改竄防止属性とする例を示したが、例えば人員管理データベースにおいて、姓と名を列を分けているような場合、婚姻等により姓が変更となる場合は姓の変更、すなわちデータの更新を必要とする場合が想定される。また、図 4 においては行挿入後、削除禁止期間内は一切の更新を不可とする例を示したが、挿入したデータ中にキー入力ミスなどによる誤りがあり、データの修正および更新をしたい場合が実際の運用上では少なくない想定される。ここで、S Q L 1 は表 4 の改竄星属性有りの列に対する更新を指示する S Q L、S Q L 2 は表 4 の改竄防止属性無しの列に対する更新を指示する S Q L である。行削除 S Q L 実行時の流れについては、図 4 と同じであるため説明は省略する。改竄防止属性有りの表 4 に対し、改竄防止属性有りの列に対する表データ更新を指示する S Q L ( S Q L 1 ) を実行した場合、データベース管理システムは更新対象となる行が挿入日当日であれば U P D A T E 可、挿入日当日でなければ不可の旨を示す情報を U A P に返す ( 1 2 0 1 )。改竄防止属性有りの表 4 に対し、改竄防止属性有りの列に対する表データ更新を指示する S Q L ( S Q L 2 ) を実行した場合、データベース管理システムは U P D A T E 可の旨を示す情報を U A P に返す。 ( 1 2 0 1 )

図 1 3 は、図 5 と同様に表データの更新処理の流れを示す図であり、列単位での改竄防止属性の有無により更新可／不可の制御を行い、かつ行挿入日当日であれば表データの更新を可能とする例を示す。図 5 においては、表データ更新要求があった場合、まず改竄防止属性有りの表か否かをチェック ( 5 0 1 ) し、更新不可 ( 5 0 6 ) としているが、行挿入当日であれば表データの更新を可能とする場合においては、改竄防止属性有りの表か否かをチェック後に更新する行の挿入日が行を挿入した当日であるかどうかのチェック ( 1 3 0 6 ) を加えることで実現できる。表データ更新要求があった場合、まず改竄防止属性有りの表か否かを

チェックし（1301）、改竄防止属性無しの場合は表データ更新実行者が表の所有者か否かチェックし（1302）、表の所有者であれば更新処理を実行する（1304）。1302で表の所有者でない場合は、表データ更新実行者が表の所有者から当該表に対する更新権限を譲渡されているかチェックし（1303）、譲渡されていれば更新処理を実行し（1304）、譲渡されていなければ更新不可とする（1305）。1301で改竄防止属性有りの場合は、更新する行の挿入日が行納入当日であるかをチェックし（1306）、行挿入当日の場合は1302以降の処理を実行する。行挿入当日でない場合は、更新する列が改竄防止属性有りの列であるかをチェックし（1307）、改竄防止属性有りの列である場合は例えば表の所有者であったとしても更新不可となる（1308）。改竄防止属性有りの列でない場合は1302以降の処理を実行する。本実施例では、改竄防止属性有りの表においての行の更新猶予期間を行挿入日当日としているが、行挿入日時保持列を持つ表であれば、行挿入日時から24時間以内や12時間以内など、詳細な更新猶予期間を設けることも可能である。この場合は、更新する行の挿入日が行挿入当日であるかどうかのチェック（1306）内容を「行挿入日時保持列を持つ日時から24時間以内か」などに変更することで実現可能である。以上、図12および図13では、列単位で改竄防止属性が設定されている場合での更新猶予期間を設ける実施例を示したが、図2に示すような列単位での改竄防止属性が設定されていない場合においては、列単位での改竄防止属性が設定がされていないことを改竄防止属性ありと見なし、図13中の更新する列が改竄防止属性有りの列であるかをチェック（1307）に、「または、改竄防止属性が設定されていない列か」という条件を加えることで実現可能である。

#### 【0028】

図14は、列単位での改竄防止属性が定義されている表への表定義の変更処理の流れを示す図である。表定義変更要求があった場合、まず改竄防止属性有りの表か否かをチェックし（1401）、改竄防止属性無しの場合は表定義変更実行者が表の所有者又はDBA権限者か否かをチェックし（1402）、表の所有者又はDBA権限者であれば表定義変更処理を実行する（1403）。ここでの「DBA権限者」とは、データベース管理システムにおける管理者権限を持つユー

ザのことである。通常、データベース管理システムにおいては、「DBA権限者」と「一般利用者」に大別され、DBA権限者は一般利用者には利用できないデータベースを管理する機能を利用することが可能である。1402で表の所有者又はDBA権限者でない場合は、表定義変更不可となる(1404)。1401で改竄防止属性有りの表の場合は、表名の変更、改竄防止属性の解除、行削除禁止期間の変更、又は行挿入日時保持列の削除のいずれかであるか否かをチェックし(1405)、1405のいずれかに合致する場合は、例えば表の所有者又はDBA権限者であったとしても表定義変更不可となる(1408)。1405でいずれの条件にも合致しない場合は、列の定義変更であるかをチェックし(1406)、列の定義である場合は改竄防止属性有りの列かをチェックし(1407)、改竄防止有りの列である場合は、例えば表の所有者又はDBA権限者であったとしても表定義変更不可となる(1408)。1407にて、改竄防止属性有りの列ではない場合は、1402以降の処理を行う。1406にて、列の定義変更で無い場合は、例えば表の所有者又はDBA権限者であったとしても表定義変更不可となる(1408)。

#### 【0029】

##### 【発明の効果】

不正にまたは誤ってデータベースへの処理が実行されることを防ぐことが可能となる。

##### 【図面の簡単な説明】

【図1】本発明に係わるデータベース管理方法及び装置である。

【図2】改竄防止を定義する定義文と改竄防止情報を格納するディクショナリ表の内容の一例である。

【図3】データ挿入時における流れを示す図である。

【図4】表データ削除SQL及び行削除SQLを実行した場合の例である。

【図5】データ更新時における流れを示す図である。

【図6】行削除時における流れを示す図である。

【図7】表定義削除時における流れを示す図である。

【図8】表定義変更時における流れを示す図である。

【図 9】 アンロード・リロード処理を実行した場合の例である。

【図 1 0】 リロード処理における流れを示す図である。

【図 1 1】 改竄防止を定義する定義文と改竄防止情報を格納するディクショナリ表の内容の一例である。

【図 1 2】 表データ更新 S Q L を実行した場合の例である。

【図 1 3】 データ更新時における流れを示す図である。

【図 1 4】 表定義変更時における流れを示す図である。

【符号の説明】

9 0 1 : アンロードユティリティ

9 0 2 : リロードユティリティ

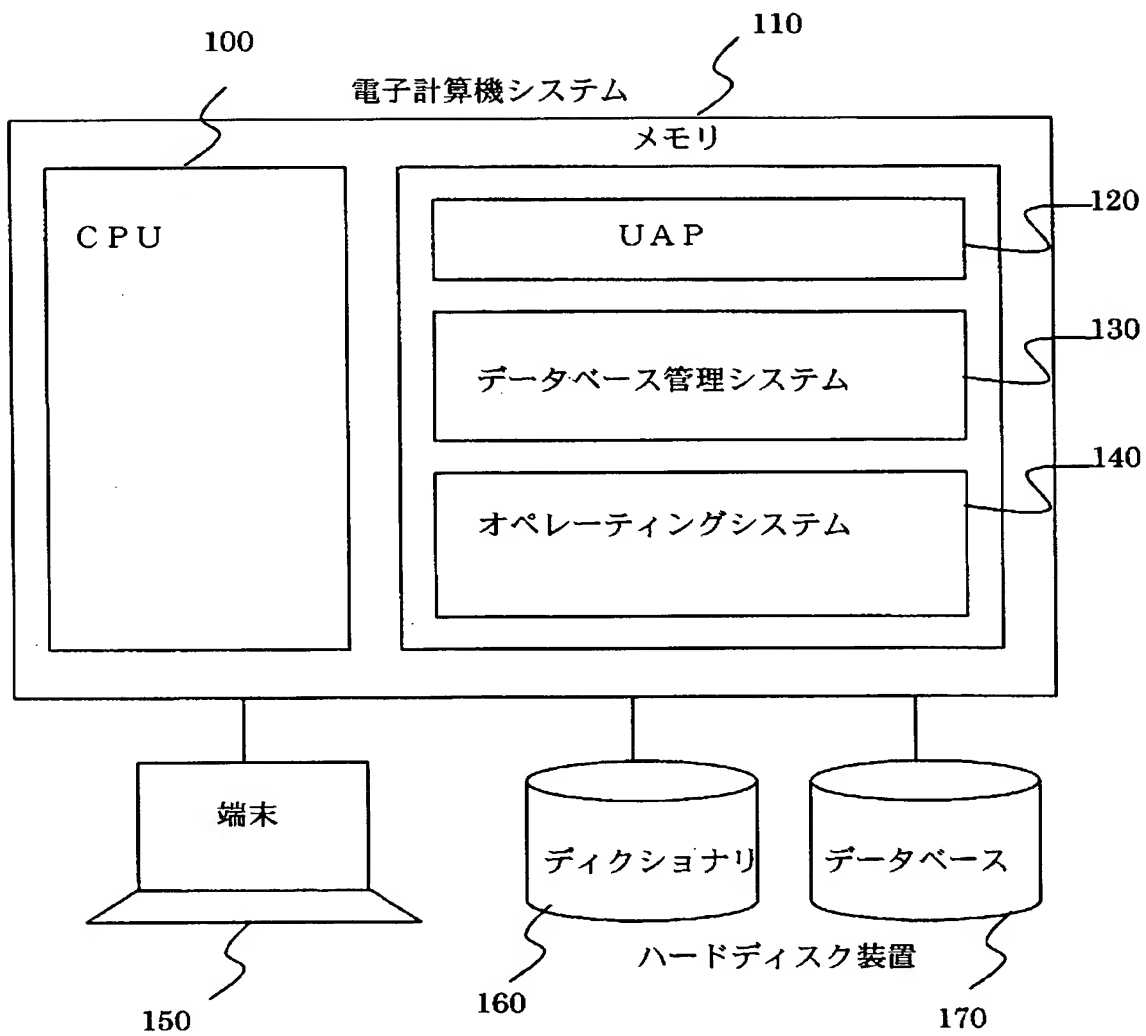
9 0 3 : 外部記憶媒体

9 0 4 : 外部記憶媒体

【書類名】 図面

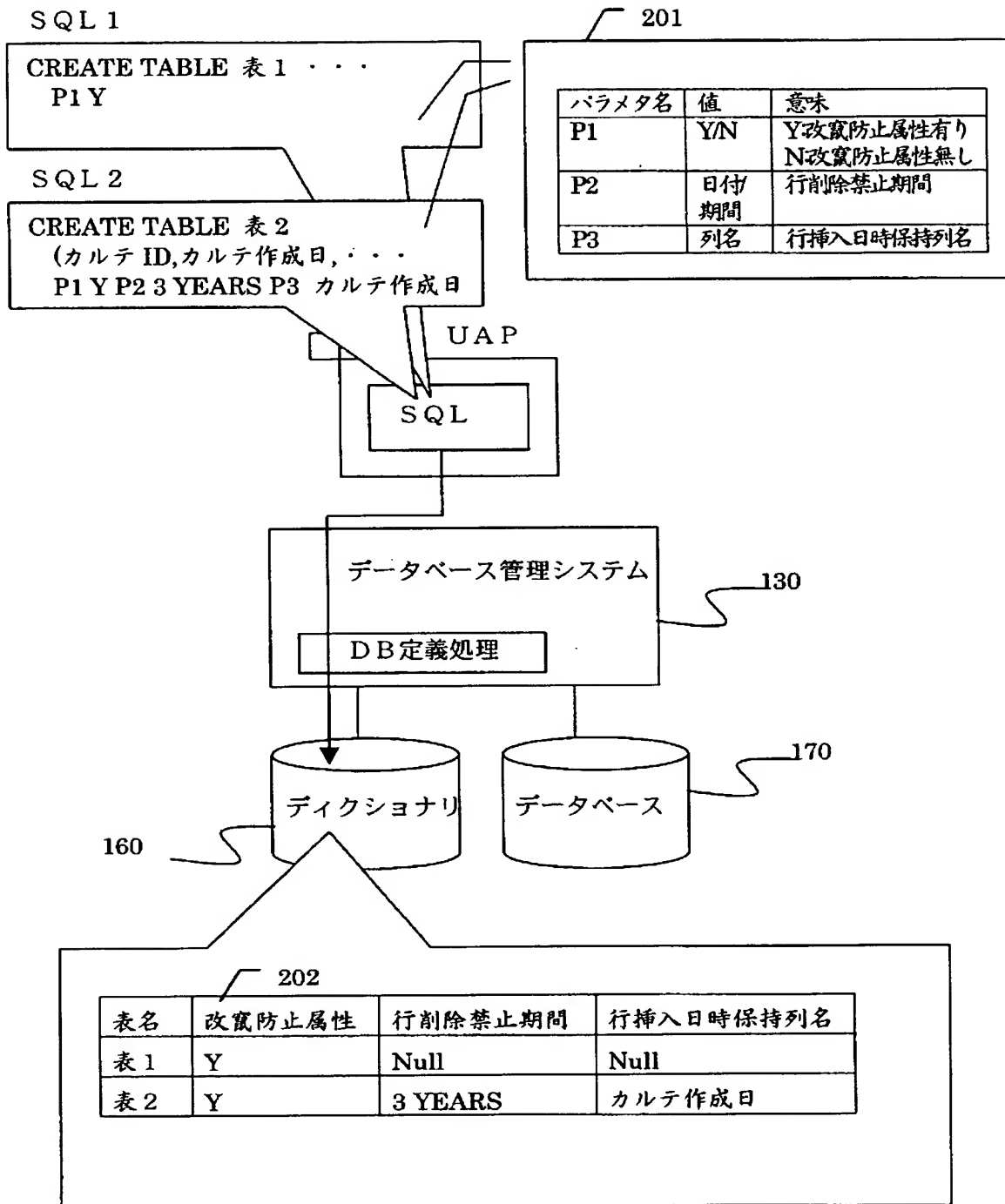
【図 1】

図 1



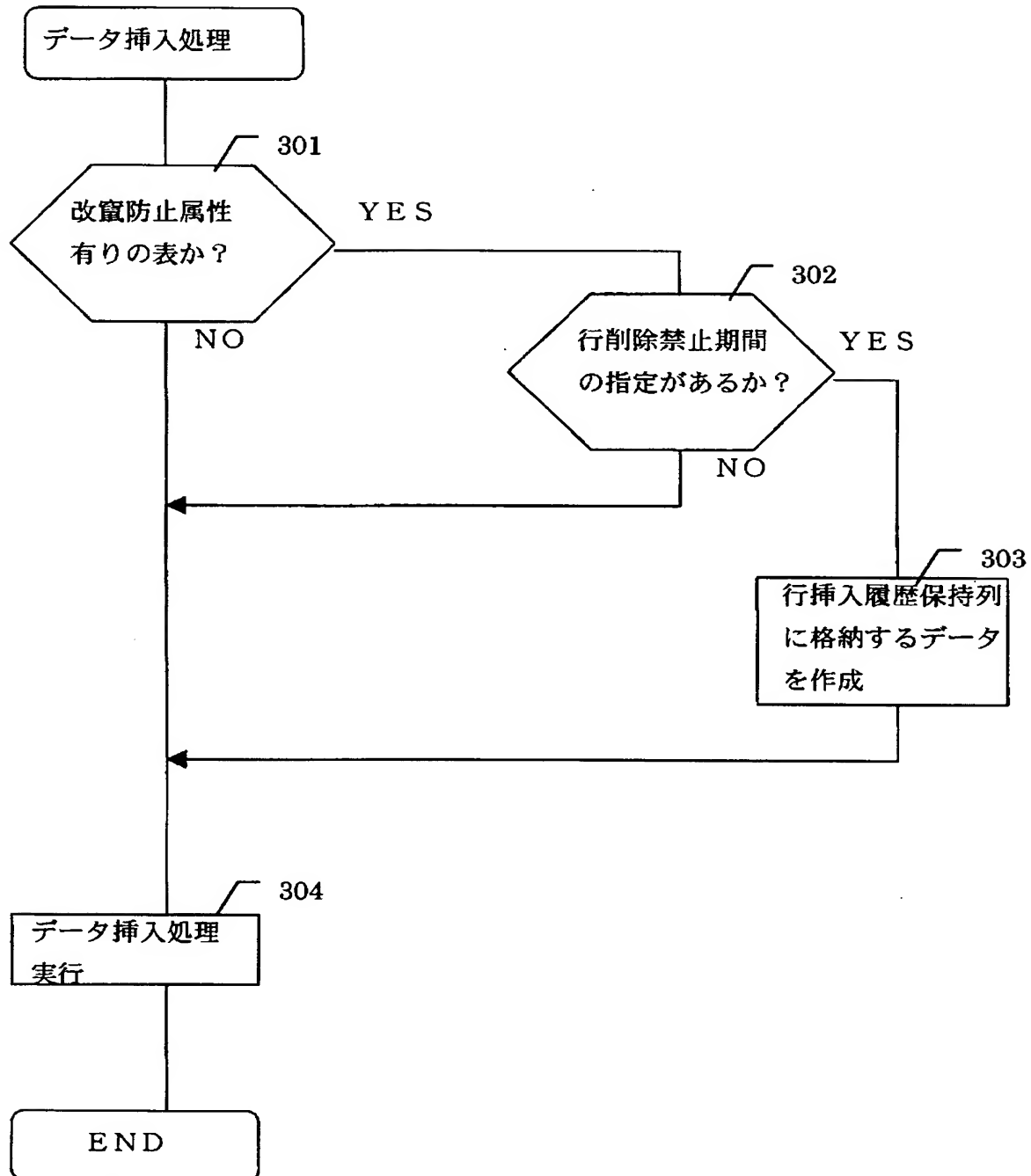
【図 2】

図 2



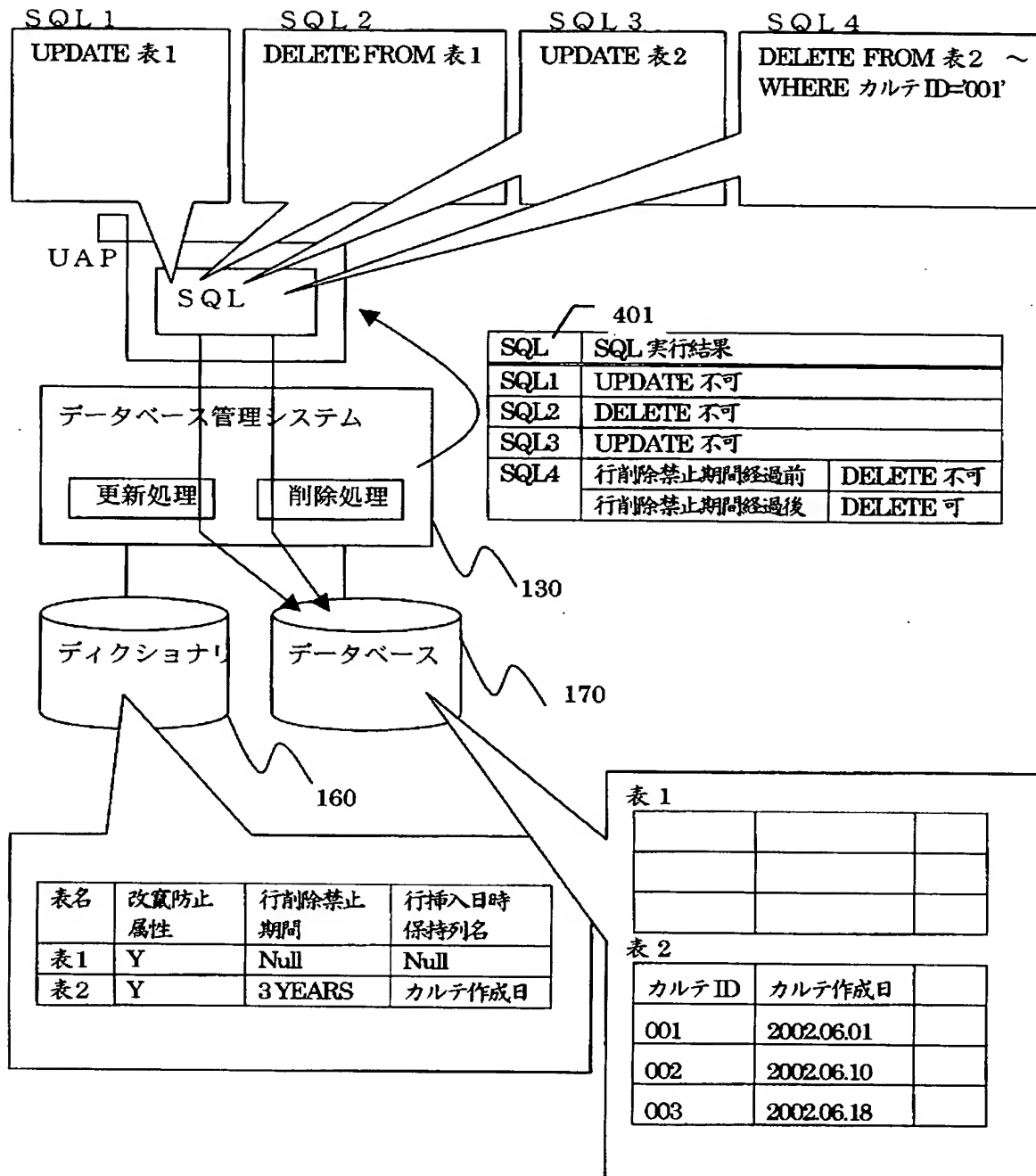
【図 3】

図 3



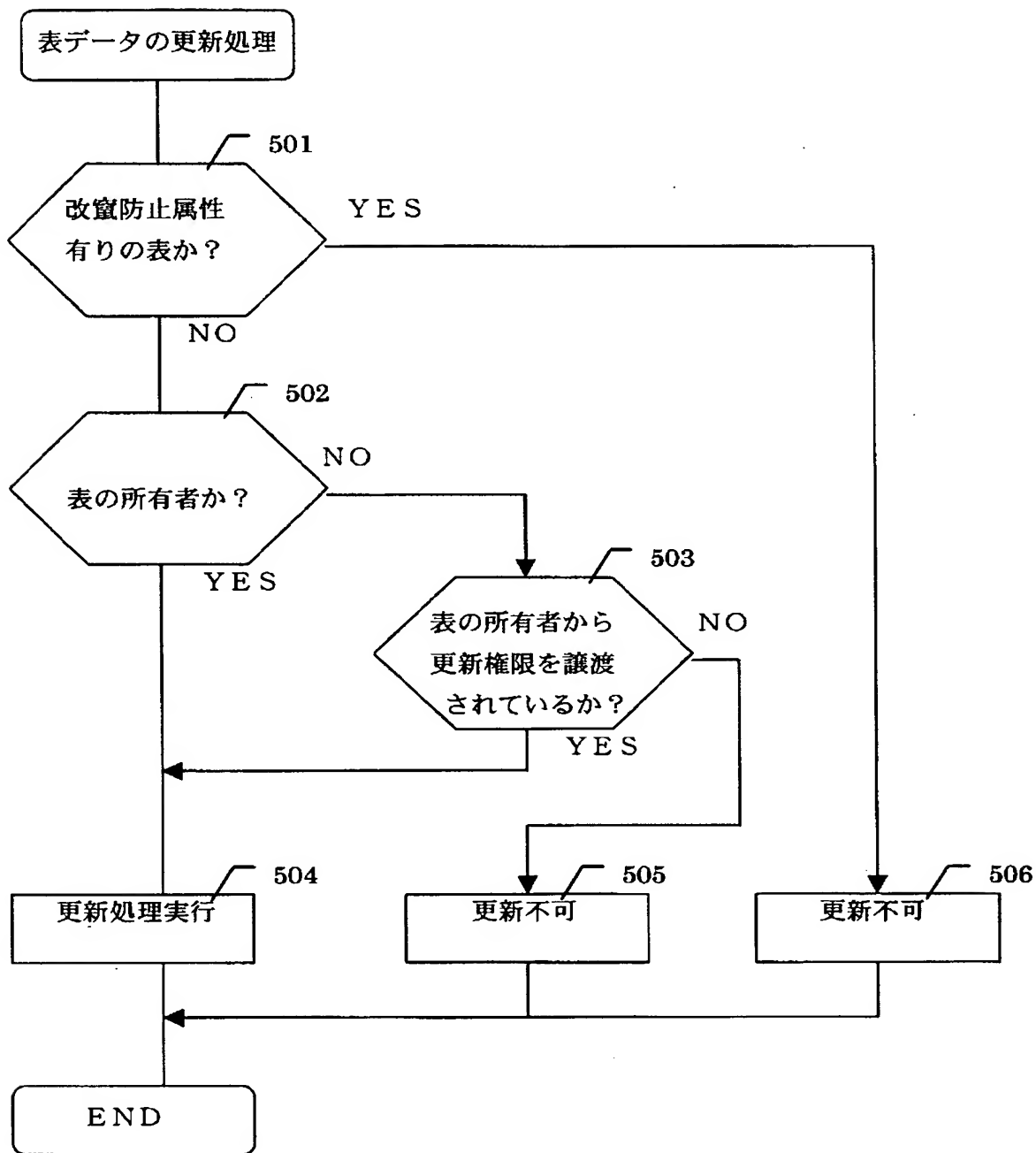
【図 4】

図 4



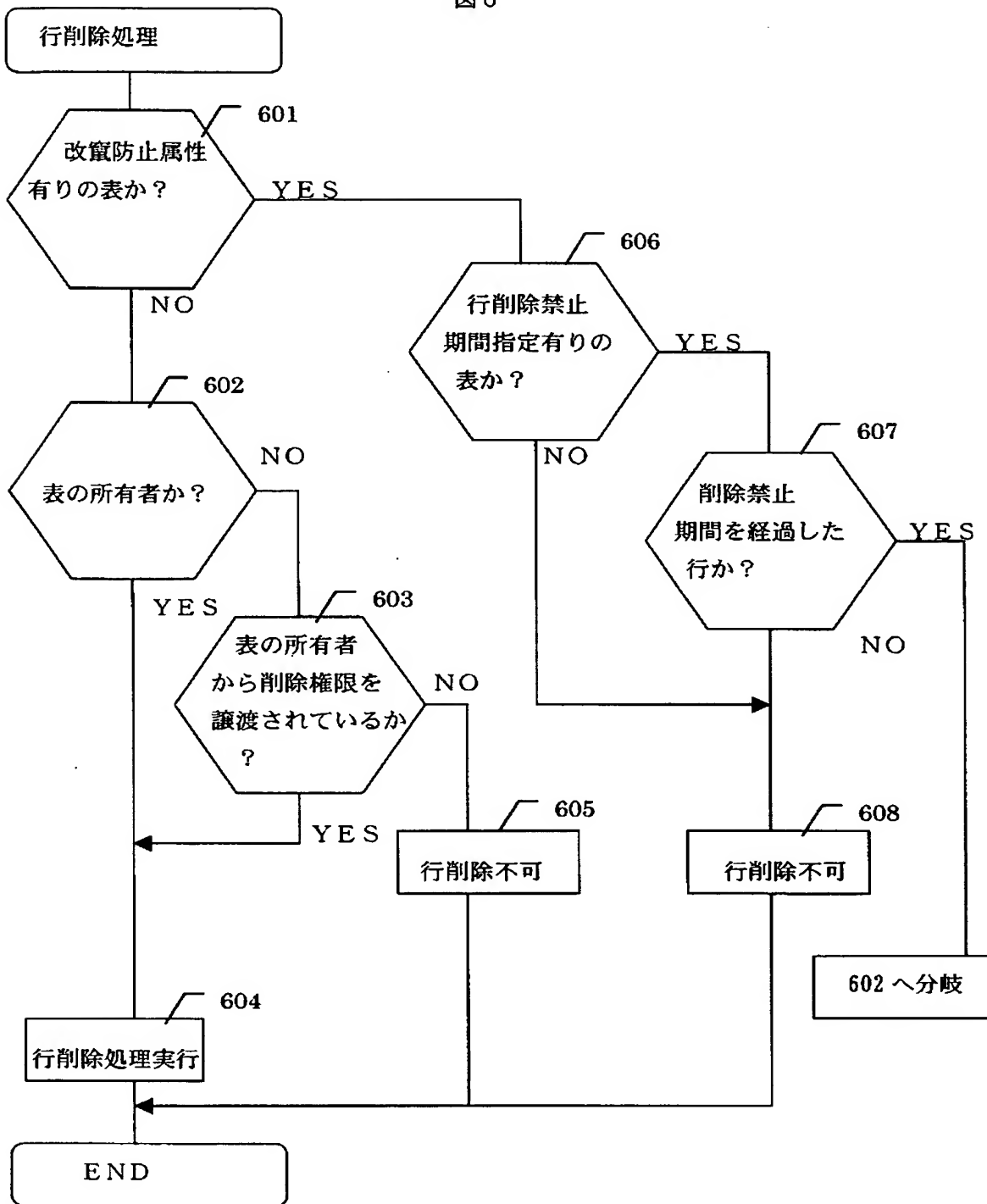
【図 5】

図 5



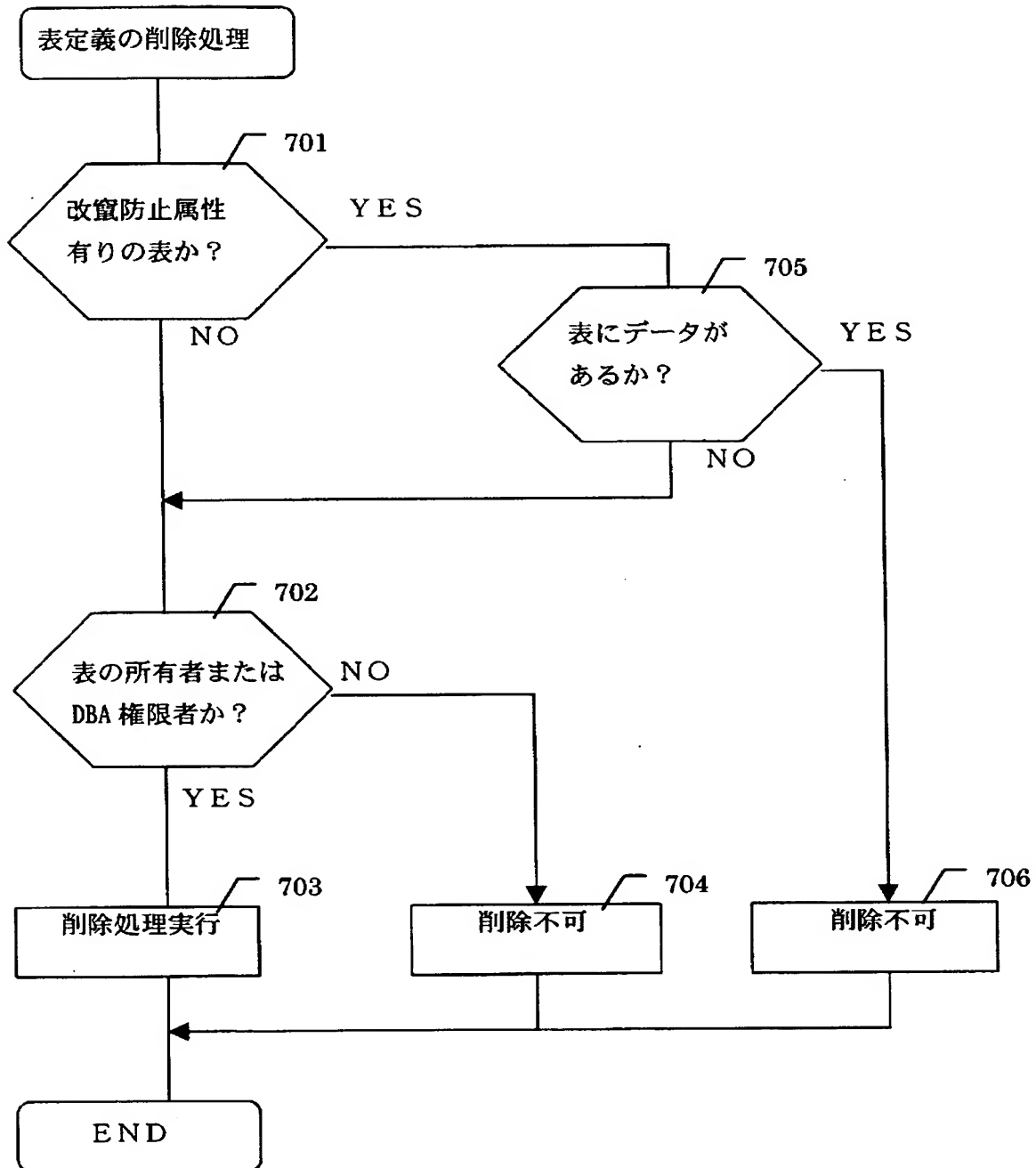
【図 6】

図 6



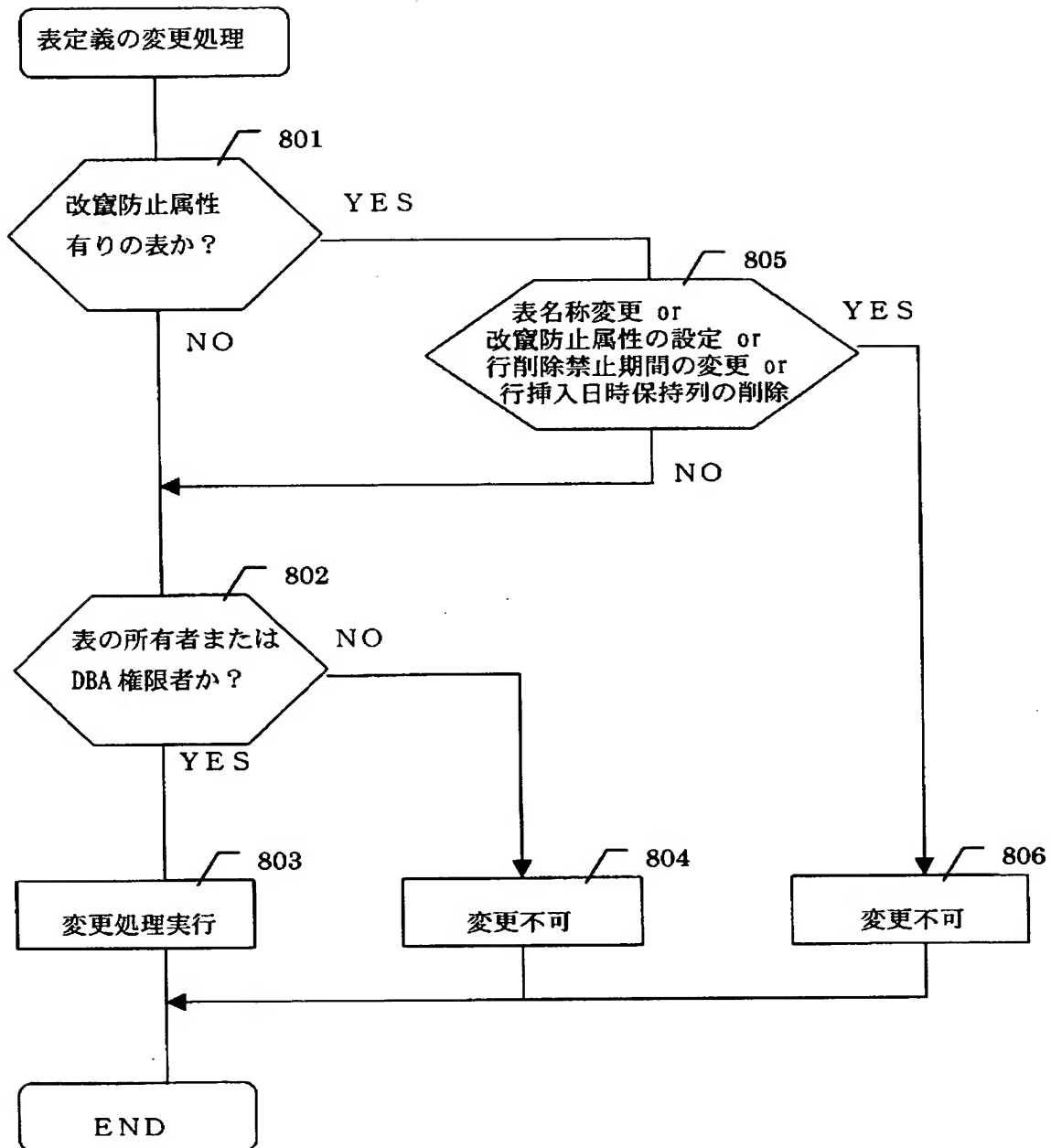
【図 7】

図 7



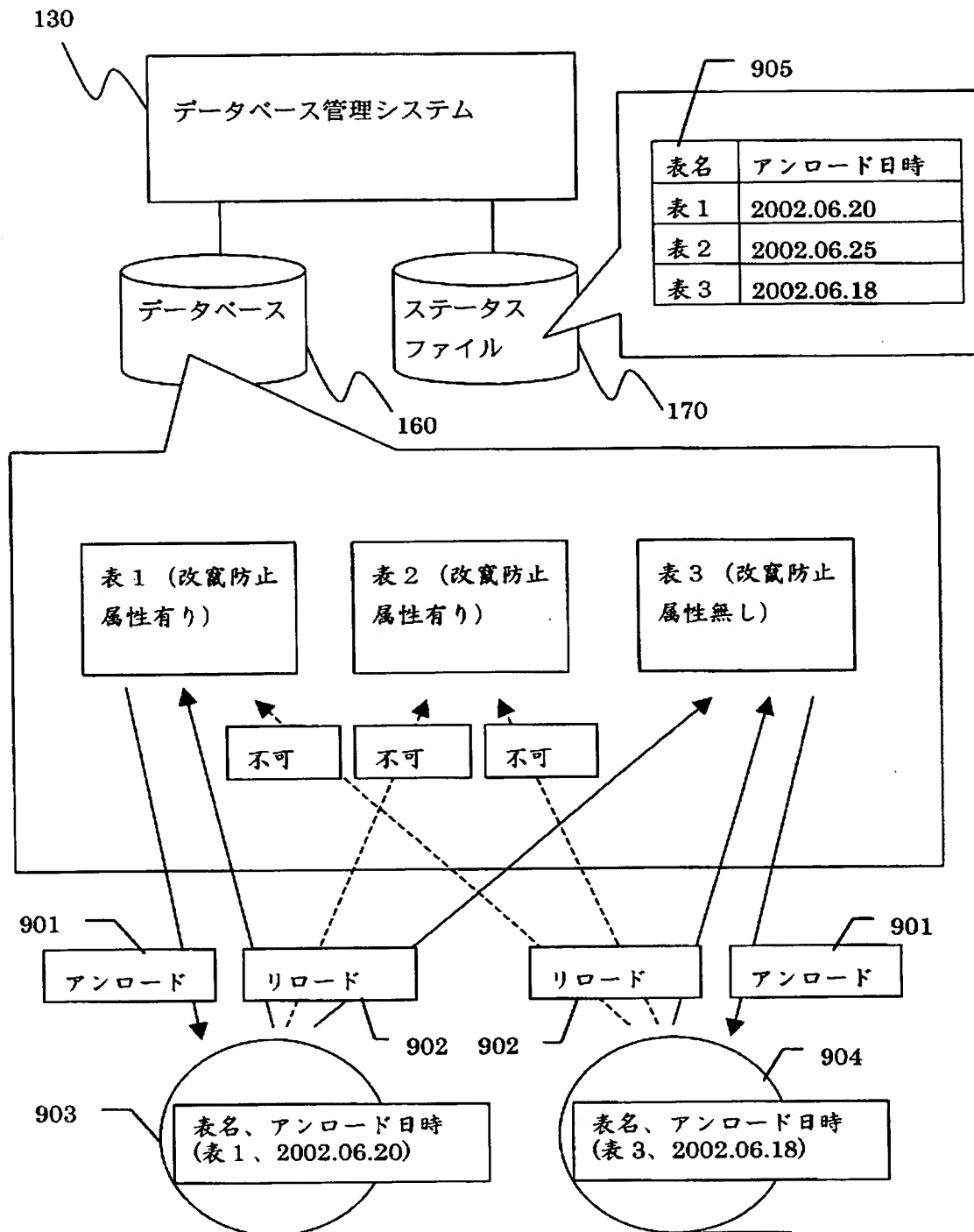
【図 8】

図 8



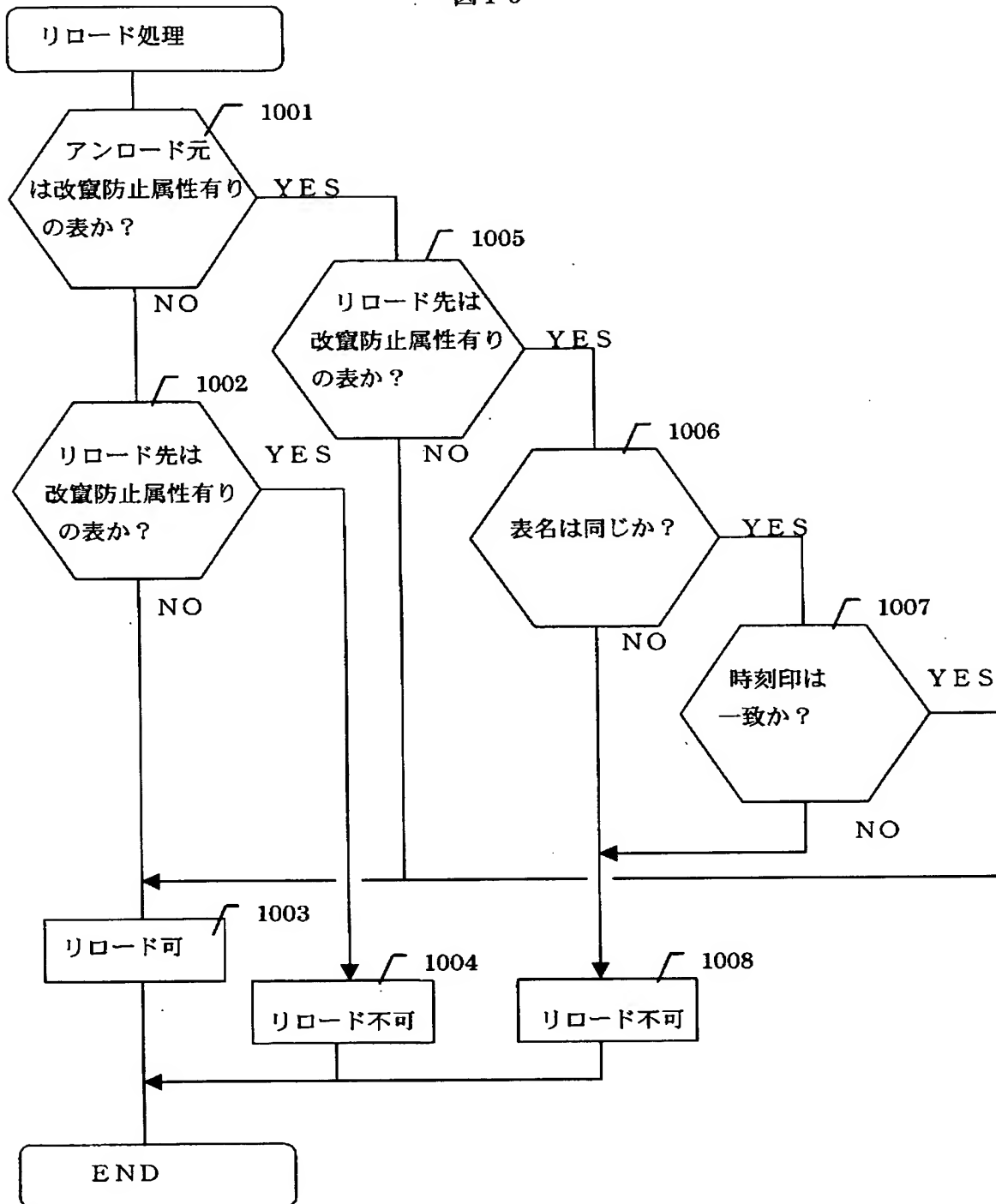
【図 9】

図 9

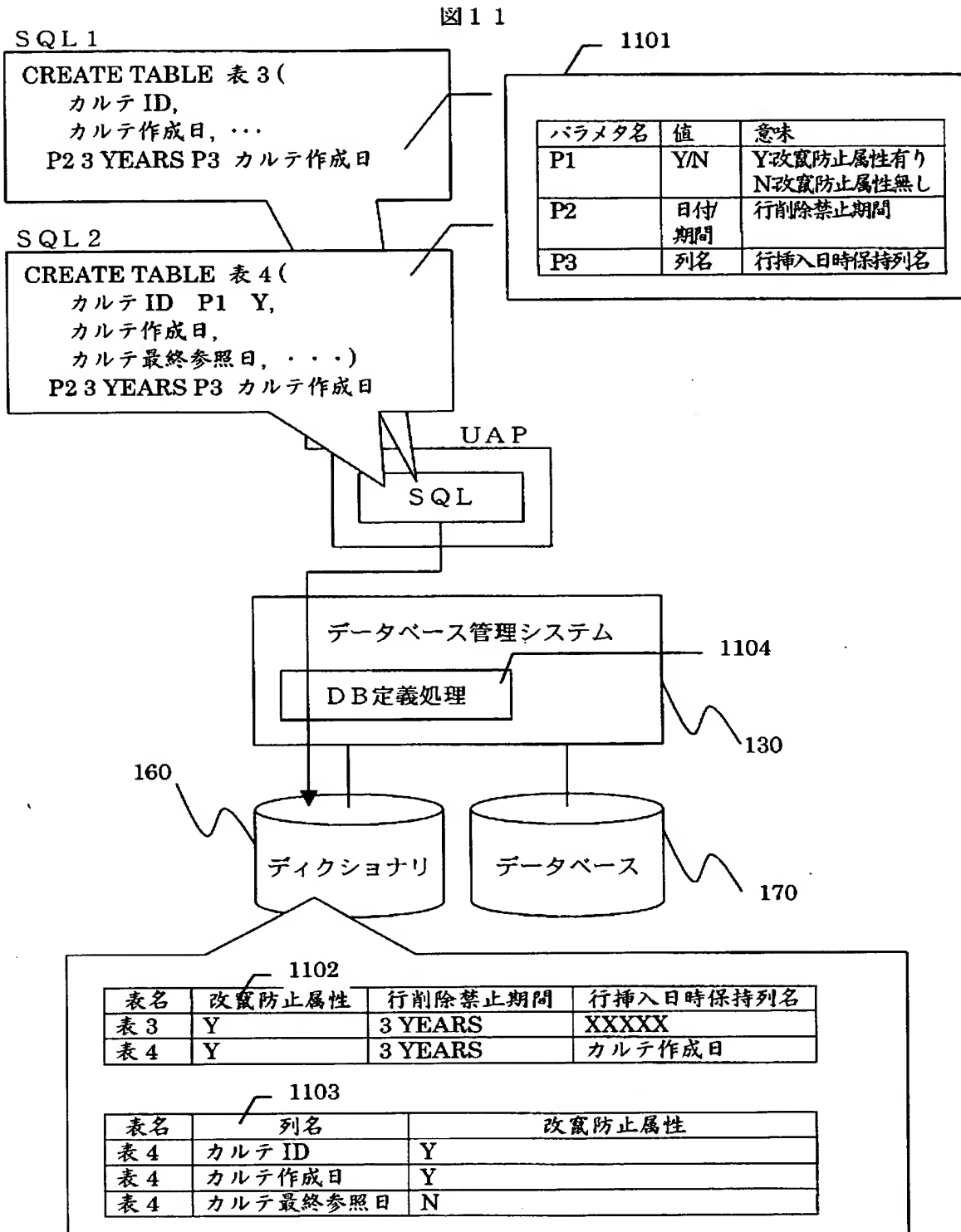


【図10】

図10

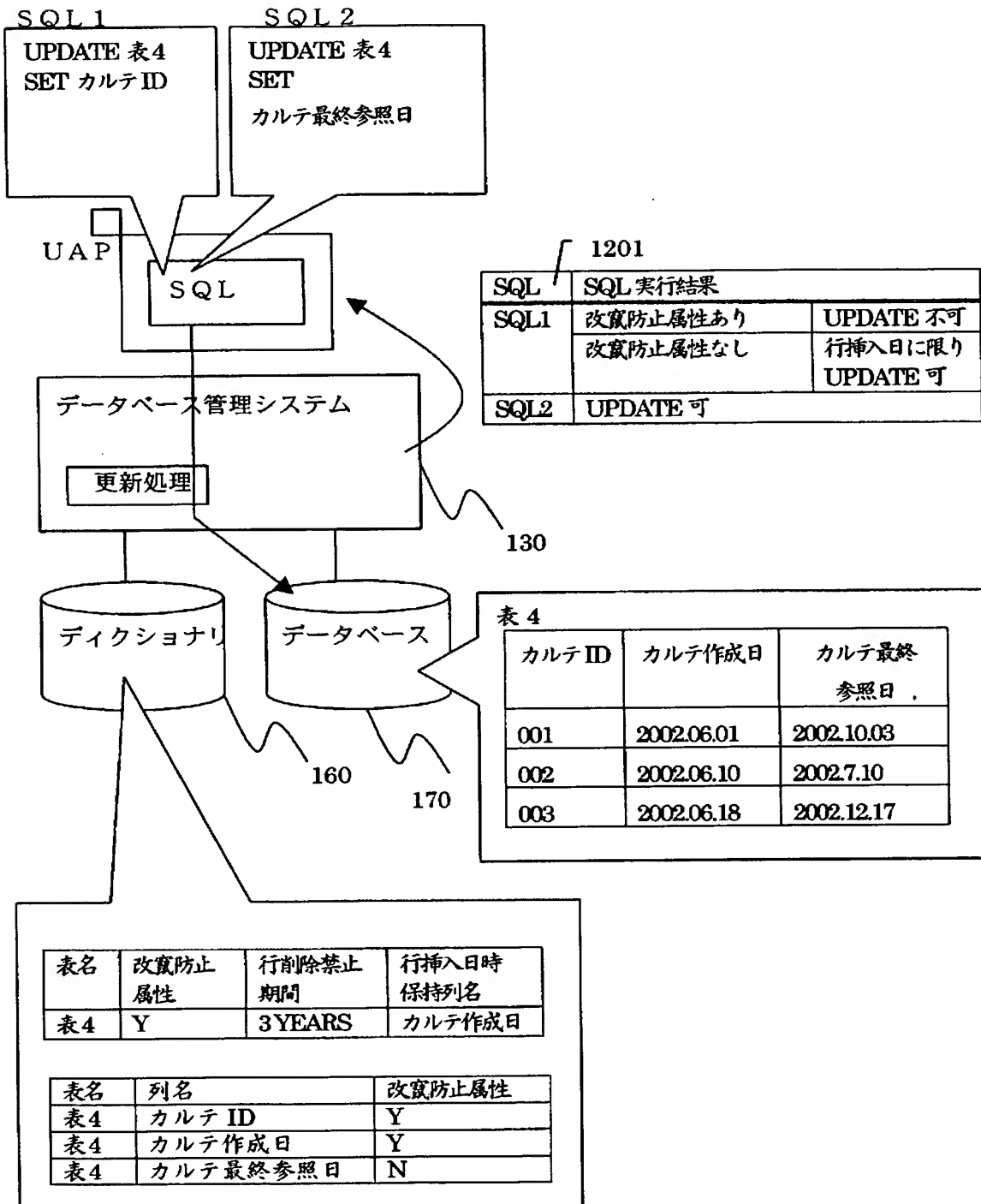


【図 11】



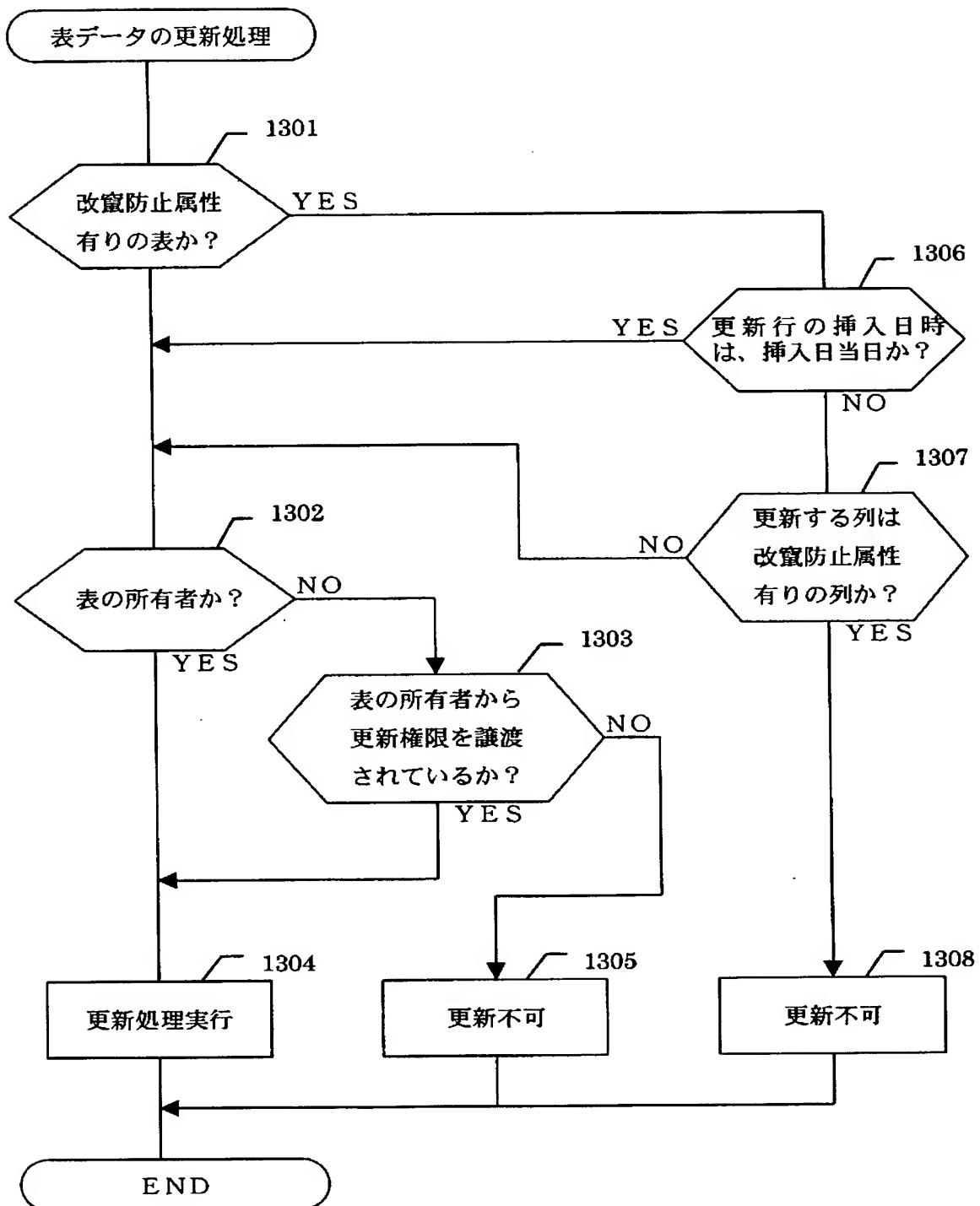
【図 12】

図 12



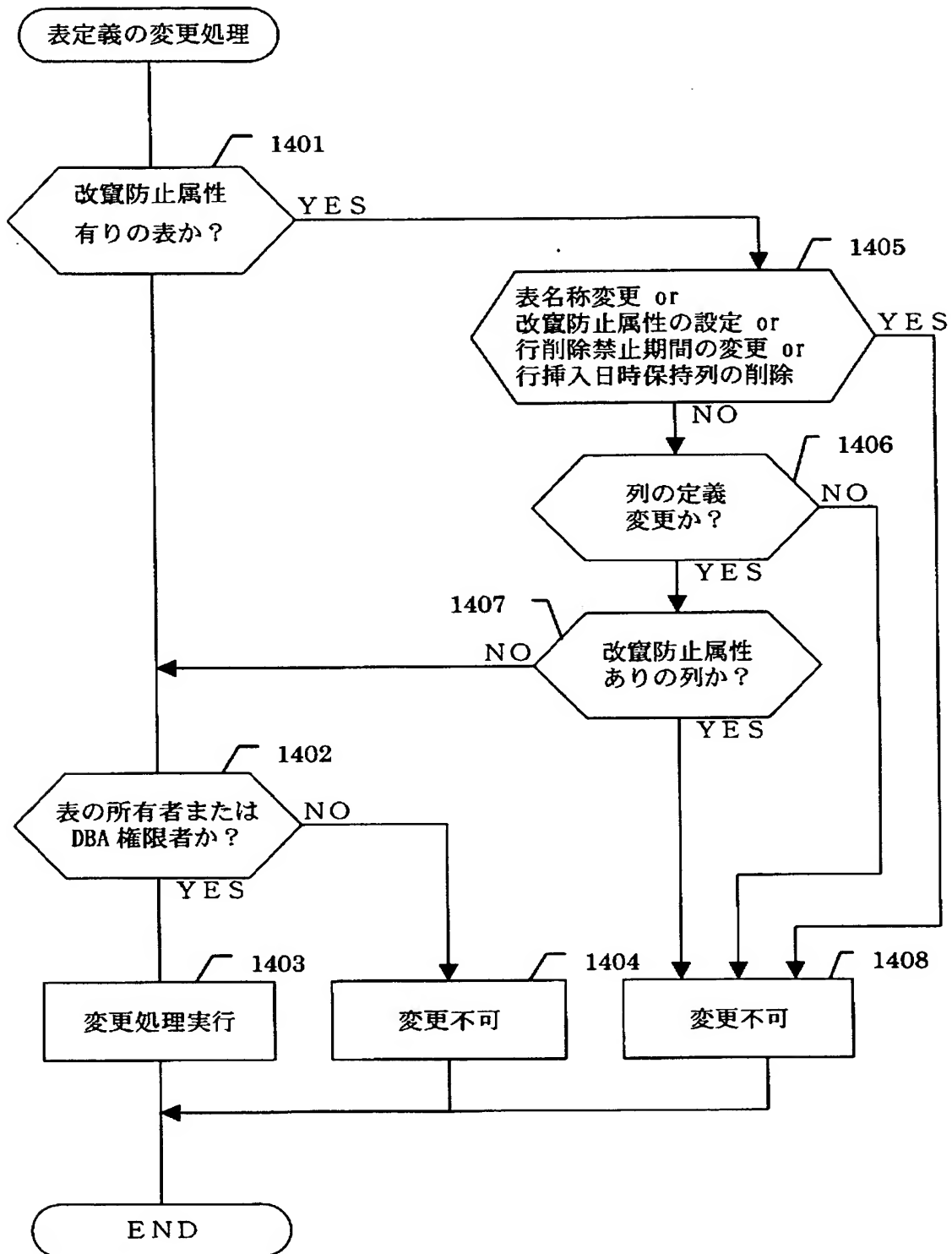
【図 13】

図 13



【図 14】

図 14



【書類名】 要約書

【要約】

【課題】

本発明はデータベース管理システムにおけるデータを不正に改竄または削除することを防ぐことにある。

【解決手段】

表の属性として改竄防止属性、行削除禁止期間、及び行挿入日時保持列名をアクセス権限属性として保持し、行挿入時に行挿入日時をデータベース管理システムが設定し、当該日時データを更新不可とする。表データ更新要求時または行削除要求時に前述表属性及び行挿入日時をチェックすることで、不正改竄または削除を防止する。

【選択図】 図 1

認定・付加情報

特許出願の番号	特願 2 0 0 3 - 1 7 7 4 5 8
受付番号	5 0 3 0 1 0 3 8 1 0 0
書類名	特許願
担当官	土井 恵子 4 2 6 4
作成日	平成 1 5 年 6 月 2 6 日

< 認定情報・付加情報 >

【提出日】 平成15年 6月23日

次頁無

特願 2003-177458

出願人履歴情報

識別番号

[000005108]

1. 変更年月日

1990年 8月31日

[変更理由]

新規登録

住所

東京都千代田区神田駿河台4丁目6番地

氏名

株式会社日立製作所

特願 2 0 0 3 - 1 7 7 4 5 8

出 願 人 履 歴 情 報

識別番号

[ 3 9 1 0 0 2 3 9 5 ]

1. 変更年月日

1 9 9 0 年 1 2 月 1 9 日

[変更理由]

新規登録

住 所

広島県広島市中区銀山町 3 番 1 号 (ひろしまハイビル 2 1)

氏 名

日立中国ソフトウェア株式会社